

MÓDULO 3: ESTRUCTURAS DE CONTROL DE FLUJO

Eduard Lara

INDICE

1. Estructuras condicionales

1. if-else

2. switch-case

2. Estructuras repetitivas

1. for

2. while

3. do-while

4. foreach

1. ESTRUCTURAS CONTROL FLUJO

- ❖ Los programas que desarrollamos no tienen porque ser sencillamente un conjunto de líneas de código fuente que se ejecutan una detrás de otra.
- ❖ A veces es necesario expresar condiciones booleanas (verdadero o falso) para decidir que líneas de código ejecutar, o sencillamente ejecutar una determinada lógica de forma reiterada.
- ❖ Las estructuras de control de flujo son las estructuras que permiten controlar el flujo de ejecución de un programa.
- ❖ Por ejemplo si queremos recuperar los valores asignados dentro de una matriz tendremos que recorrer todos sus elementos con una estructura de control que nos permita acceder de uno en uno.
- ❖ Otro ejemplo es el de una estructura que nos permite decidir según unos determinados valores obtenidos para las variables que controlamos ejecutar una línea de programa u otra dependiendo del valor analizado.

1. ESTRUCTURAS CONDICIONALES

IF - ELSE

- ⊙ Estructura selectiva **if - else**

```
if (condición)
    sentencia
```

```
if (condición)
    sentencia 1
else
    sentencia 2
```

```
if (condición1)
    sentencia 1
else if (condición2)
    sentencia 2
...
else if (condición n)
    sentencia n
else
    sentencia n+1
```

- ⊙ Mismo comportamiento que en C
- ⊙ Las sentencias compuestas se encierran entre llaves
- ⊙ elseif puede ir todo junto

1. ESTRUCTURAS CONDICIONALES

IF - ELSE

Hacemos una pregunta compuesta para una expresión booleana y en caso de ser cierto se ejecutará el código fuente alojado dentro de la sentencia if

```
13  -
14
15  -
16
17
18  -
```

```
<?php
    $edad = 19;
    if($edad >= 18) {
        echo "És major d'edat";
    }
?>
```

Si, en efecto, la variable edad contiene un valor superior o igual a 18, se ejecutará la línea de código fuente 16

1. ESTRUCTURAS CONDICIONALES

IF - ELSE

Estructura if-else intercalada dentro de instrucciones PHP	
<pre><?php if (condicion1) { código si se cumple condición1 } else { código sino se cumple condición1 } ?></pre>	<pre><?php if(condicion1){ instrucciones1; } elseif (condicion2) { instrucciones2; } else { instrucciones3; } ?></pre>

1. ESTRUCTURAS CONDICIONALES

IF - ELSE

Estructura if-else intercalada dentro de
instrucciones HTML

```
<?php if(condicion1) {?>  
    instrucciones HTML1 si se cumple condicion1  
<?php } else{  ?>  
    instrucciones HTML2 si no se cumple condicion1  
<?php } ?>
```

1. ESTRUCTURAS CONDICIONALES

IF - ELSE

Ejemplo de estructura selectiva **if-else**:

```
<?PHP
    if ($sexo == 'M')
        $saludo = "Bienvenida, ";
    else
        $saludo = "Bienvenido, ";
    $saludo = $saludo . $nombre;
    print ($saludo);
?>
```



1. ESTRUCTURAS CONDICIONALES

IF - ELSE

Ejemplo de estructura selectiva **if-else**:

```
<?php
    $dia_semana = date("w");
    // 0=Domingo,... 6=Sábado
    if ($dia_semana == 4){ //Representación numérica de Jueves
        echo "Hoy es Jueves!!";
    }else
        echo "Hoy no es Jueves!!";
?>
```

1. ESTRUCTURAS CONDICIONALES

IF - ELSE

En cas de que la condició resulti en **false** , podem també definir altra lògica gràcies a la sentència **else**

```
13  <?php
14  $edad = 15;
15  if($edad >= 18) {
16      echo "És major d'edat";
17  }
18  else {
19      echo "És menor d'edat";
20  }
21  ?>
```

En aquest cas, al ser el valor 15, s'executarà la lògica de la línia 19. Si no estigués definida la sentència **else**, senzillament no s'executaria res, en aquest cas, al evaluar-se la condició

1. ESTRUCTURAS CONDICIONALES

IF - ELSE

Si hi ha més casos que volem tenir en compte per a evaluar una condició concreta, podem fer servir **else if**

```
<?php
    $edad = "";
    if($edad >= 18) {
        echo "És major d'edat";
    }
    else if(!is_numeric($edad)) {
        echo "No és un número!";
    }
    else {
        echo "És menor d'edat";
    }
?>
```

En aquest cas, també comprovem que no sigui un número. En cas de ser així, es mostrarà el missatge:

“No és un número”

1. ESTRUCTURAS CONDICIONALES

SWITCH - CASE

Estructura selectiva **switch**

```
switch ( variable ) {  
    case valor1:  
        sentencia 1;  
        break;  
    case valor2:  
        sentencia 2;  
        break;  
    default:  
        instrucciones en caso de no cumplirse ninguna condición  
}
```

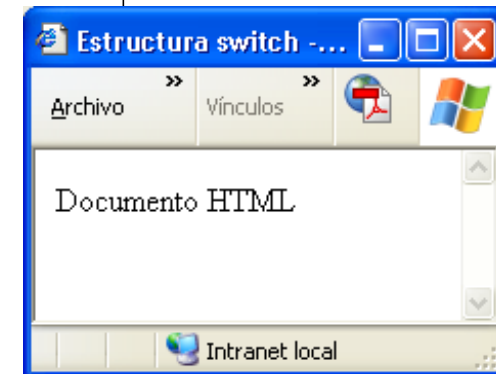
Ideal para evaluar numerosas condiciones.
Mismo comportamiento que en C, sólo
que la expresión del case puede ser
integer, float o string

1. ESTRUCTURAS CONDICIONALES

SWITCH - CASE

Ejemplo de estructura selectiva **switch**:

```
switch ($extension) {  
    case ("TXT"):   
        $tipo = "Documento de texto";  
        break;  
    case ("HTML"):   
    case ("HTM"):   
        $tipo = "Documento HTML";  
        break;  
    default:   
        $tipo = "Archivo " . $extension;  
}  
print ($tipo);
```



1. ESTRUCTURAS CONDICIONALES

SWITCH - CASE

Ejemplo de estructura selectiva **switch**:

```
<?php $a=3;
    switch($a) {
        case 1: echo "estamos en la opción uno";
                break;
        case 2: echo "estamos en la opción dos";
                break;
        case 3: echo "estamos en la opción tres";
                $a--;
                break;
        default: echo "No hay opciones";
    }
    echo "Valor de a" . $a;
?>
```

1. ESTRUCTURAS CONDICIONALES

SWITCH - CASE

Una altra manera d'establir condicionals en l'execució seria mitjançant la sentència switch

```
<?php
$favcolor = "red";

switch ($favcolor) {
    case "red":
        echo "Your favorite color is red!";
        break;
    case "blue":
        echo "Your favorite color is blue!";
        break;
    case "green":
        echo "Your favorite color is green!";
        break;
    default:
        echo "Your favorite color is neither red, blue, nor green!";
}
?>
```

S'evalua el valor de la variable \$favcolor, i, en funció d'aquesta evaluació s'efectuarà només el cas que coincideixi.

Important la instrucció break per no continuar executant el condicional quan ja s'ha executat un dels casos.

El cas default és equivalent a else.

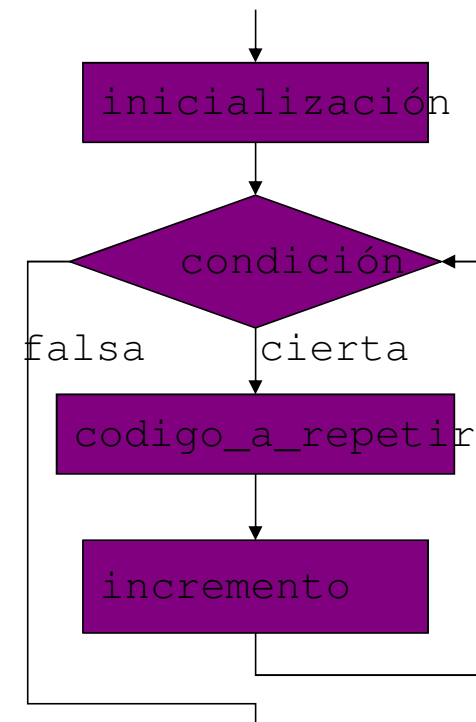
2. ESTRUCTURAS REPETITIVAS

FOR

- ⊙ Estructura repetitiva **for**

```
for (inicialización; condición; incremento)
{
    código_a_repetir;
}
```

- ⊙ Mismo comportamiento que en C



2. ESTRUCTURAS REPETITIVAS

FOR

Ejemplos de estructura repetitiva **for**:

```
<?php  
    print("<UL>\n");  
    for ($i=1; $i<=5; $i++)  
        print("<LI>Elemento $i</LI>\n");  
    print("</UL>\n");  
?>
```



2. ESTRUCTURAS REPETITIVAS

FOR

Tenim altres estructures de repetició, com aquí amb el bucle **for**

```
<?php
for ($x = 0; $x <= 10; $x++) {
    echo "The number is: $x <br>";
}
?>
```

Aquí es defineixen més elements de bucle entre parèntesis deixant només la lògica a executar dins del bucle

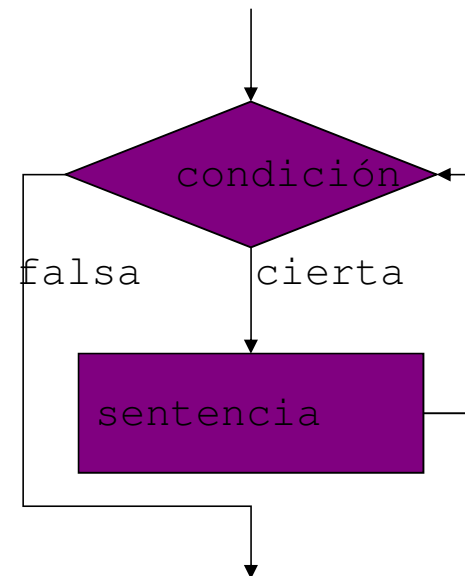
2. ESTRUCTURAS REPETITIVAS

WHILE

- ⊙ Estructura repetitiva **while**

```
while (condición)
{
    sentencia
}
```

- ⊙ Mismo comportamiento que en C



2. ESTRUCTURAS REPETITIVAS

WHILE

Ejemplo de estructura repetitiva **while**:

```
<?php
    print("<UL>\n");
    $i=1;
    while ($i <= 5)
    {
        print("<LI>Elemento $i</LI>\n");
        $i++;
    }
    print("</UL>\n");
?>
```



2. ESTRUCTURAS REPETITIVAS

WHILE

En ocasions ens interessa repetir determinades instruccions i per això tenim els **bucles** o **estructures de repetició**

```
<?php
$x = 1;

while($x <= 5) {
    echo "The number is: $x <br>";
    $x++;
}
?>
```

Aquesta és la condició que comprovem **a cada iteració del bucle**

Aquí per exemple, escrivim un missatge amb número de 1 al 5 que es calcula en cada iteiteració del bucle

A cada iteració del bucle, executem la lògica especificada dins del mateix

IMPORTANT: Hem d'establir **SEMPRE** la lògica que trencarà el bucle, o sino, deixarem l'execució del programa penjada en un **bucle infinit**. Aquí, per exemple, la línia **\$x++** s'encarrega de que, després de 5 iteracio₇ns, el bucle es trenqui

2. ESTRUCTURAS REPETITIVAS

DO-WHILE

Bucle (while)	Ejemplo
<pre>do { codigo a repetir } while</pre>	<pre><?php \$valor1=3; \$valor2=0; do { echo \$valor1." x" . \$valor2 . "="; echo \$valor1 *\$valor2; \$valor2++; } while(\$valor2>=10); ?></pre>

2. ESTRUCTURAS REPETITIVAS

DO-WHILE

Menys comú, però existent al cap i a la fi

```
<?php
$x = 1;

do {
    echo "The number is: $x <br>";
    $x++;
} while ($x <= 5);
?>
```

La principal diferència respecte a les altres estructures de repetició és que aquí, la comprovació de la condició que trenca el bucle es fa **al final** d'una iteració

2. ESTRUCTURAS REPETITIVAS

FOREACH

- ⊙ La estructura de control **foreach** permite iterar sobre arrays, tanto indexados por claves numéricas positivas o por cadenas.
- ⊙ Sintaxis:
 - foreach (expresión_array as \$valor)
 sentencia
 - foreach (expresión_array as \$clave => \$valor)
 sentencia

2. ESTRUCTURAS REPETITIVAS

FOREACH

- ⦿ Ejemplos:

```
$color = array ('rojo'=>101, 'verde'=>51, 'azul'=>255);  
foreach ($color as $valor)  
    print "Valor: $valor<BR>\n";  
foreach ($color as $clave => $valor)  
    print "Clave: $clave; Valor: $valor<BR>\n";
```
- ⦿ Salida:

```
Valor: 101  
Valor: 51  
Valor: 255  
Clave: rojo; Valor: 101  
Clave: verde; Valor: 51  
Clave: azul; Valor: 255
```